JOURNAL OF COMPUTER AND SYSTEM SCIENCES 32, 363–369 (1986)

Uniform Normal Form for General Time-Bounded Complexity Classes

BERNARD R. HODGSON*

Département de mathématiques, statistique et actuariat, Université Laval, Québec, Québec, Canada G1K 7P4

AND

CLEMENT F. KENT*

Department of Mathematical Sciences, Lakehead University, Thunder Bay, Ontario, Canada P7B 5E1

Received August 23, 1983; revised August 28, 1985

Refining techniques of previous works, we obtain a normal form arithmetical representation for non-deterministic computability, in which the polynomial matrix does not involve the time-bounding function. This permits arithmetization of Turing machine complexity classes determined by quite general time bounds. Applications are made to complexity hierarchies and to obtain a single, uniform, normal form.

1. INTRODUCTION

In [1, 2], the authors established an arithmetical formula scheme which characterizes non-deterministic polynomial-time computability, and then improved the form by establishing limits on the number of quantifier alternations in its prefix. The resulting "normal form" for NP sets was used to interrelate several possible hierarchies of interest in the theoretical computer science literature. The authors had an opportunity to discuss these results with Yuri Matijasevič in October 1982 and are indebted to him for observing a simplification of the proof of the normal form of [2]. In exploiting Matijasevič's suggestion, a more general representation scheme emerged, which is given in Theorem 1 below. Subsequently Matijasevič has sent us [3], which contains a closely related version of Theorem 1 obtained independently by Jukna. We comment on the relation of the two results below.

Section 2 summarizes and sketches the improvements needed for the new normal form, and Section 3 presents applications to hierarchies and uniform representability. Readers are referred to [1, 2] for background.

* The authors wish to acknowledge the partial support of the Natural Sciences and Engineering Research Council of Canada, under Grants A4519 and A5603, and partial support of the Ministries of Education of the Provinces of Québec and Ontario under the Ontario-Québec Exchange Program.

0022-0000/86 $3.00

2. Improved Normal Form for Polynomially Closed Classes of Time Bounds

The normal form of [2] is an arithmetical expression

$$\bigvee_{y}^{2^{p(|\mathbf{n}|)}} \bigwedge_{u}^{q(|\mathbf{n}|)} \bigvee_{v_1}^{2^{r_1(|\mathbf{n}|)}} \cdots \bigvee_{v_m}^{2^{r_m(|\mathbf{n}|)}} [Q_R(\mathbf{n}, y, u, v_1, \ldots, v_m) = 0] \tag{1}$$

which characterizes membership of l-tuples $\mathbf{n} = (n_1, \ldots, n_l)$ of natural numbers in an NP-acceptable relation R (i.e., $\mathbf{n} \in R \Leftrightarrow (1)$). The quantifier bounds p, q, $r_1, \ldots, r_m$ are polynomial functions, with positive coefficients, of the l arguments $|\mathbf{n}| = (|n_1|, \ldots, |n_l|)$, and Q_R is a polynomial, with integer coefficients, which depends, as does m, on the Turing machine which accepts R. Further, the time-bounding polynomial, $\beta(\mathbf{x})$, forms a part of Q_R in our previous proof. In [1, 2], such forms were called EEBA (exponential existential bounded arithmetical) and are of quantifier prefix form $\sum_3^0$. We sometimes write $\sum_3^0$—EEBA to stress both the bounds and the prefix.

Examination of the proofs in [1, 2] shows that the quantifier-bounding functions p, q, $r_1, \ldots, r_m$ are simple polynomial functions of the time-bounding function $\beta(\mathbf{x})$ (in the case of NP, $\beta(\mathbf{x})$ is also polynomial) and the l number arguments $|n_1|, \ldots, |n_l|$. To generalize our results to other time bounds, we summarize the needed properties of a class of such bounds in a definition.

Definition 1. B is a *polynomially closed* class of time bounds if B is a class of recursive functions (of various numbers of arguments) such that:

(i) $\beta(\mathbf{x}) \geqslant x_i$ for $1 \leqslant i \leqslant l$, $\beta \in B$ and $\mathbf{x} = (x_1, \ldots, x_l)$.

(ii) (Polynomial closure) If $\beta_1, \ldots, \beta_e$ are members of B with the same arity and p is any polynomial in e arguments, there is another member $\gamma \in B$ so that, for all $\mathbf{x}$,

$$p(\beta_1(\mathbf{x}), \ldots, \beta_e(\mathbf{x})) \leqslant \gamma(\mathbf{x}).$$

(iii) We have a set of "time constructible" indices for the elements of B so that, from index c for β, we can compute $\beta(\mathbf{x})$ in time $O(\beta(\mathbf{x}))$. (This condition is only needed in Theorem 3, below. In fact, we just need to know that $\beta(\mathbf{x})$ is computable from c in time $O((\beta(\mathbf{x}))^k)$, for any fixed k.)

Two modifications are needed to the proof of the Normal Form theorem in [2], so that it may be applied to a polynomially closed class of time bounds, B. First, we will remove Lemma 2.3 of [2] from the proof. Second, we sketch how to avoid the entry of time bound $\beta \in B$ into the matrix of the normal form by eliminating Lemma 2.2. This second step is necessary to avoid "spoiling" the polynomial matrix if the class B contains, say, much more rapidly growing functions than polynomials.

The predicate simplified by Lemmas 2.2 and 2.3, in [2], is $\mathrm{ACC}_M(\mathbf{n})$, the l-placed version of the predicate in Lemma 4.6 of [1], defined by:

$$\bigvee_x \bigvee_y \bigvee_z \{\mathrm{Accept}_M(x, y, z) \wedge x = L_l(\mathbf{n}) \wedge y = P_l(\mathbf{n}) \wedge z = 2^{L_l(\mathbf{n})-1}\}. \tag{2}$$

Removing all exponential expressions from the matrix of (2) using Adleman's diophantine characterization, as done in [2], shows that "essential" $\bigwedge$-quantifiers remain in only two contexts: the quantifier

$$\bigwedge_k^{\beta(|\mathbf{n}|)}$$

in the "acceptance predicate" $\mathrm{Accept}_M(x, y, z)$ (see [1, p. 264]), and the l quantifiers

$$\bigwedge_{j_i}^{|n_i|}, \qquad 1 \leqslant i \leqslant l$$

occuring, one each, in the "position predicates," $y_i = P(n_i)$, introduced in (2.5) of [2] (see also [1, p. 266]). In the first case, the $\bigwedge$-quantifier on k mediates the legitimacy of the successive transitions of the TM, up to the total number involved. The last l $\bigwedge$-quantifiers result from encoding the input $\mathbf{n}$ by a "locator triple" (x, y, z).

In [2], we "factored" the various universals into the prefix without care, then used Lemma 2.3 to bring separated universals into a single block. Matijasevič observed that, since each conjunctive clause of (2) is $\sum_3^0$-arithmetical, all $\bigwedge$-quantifiers can be factored, simultaneously, in a single block by prenex normal form reductions possible in the pure logic. In somewhat more detail, the first conjunctive clause of (2) is $\mathrm{Accept}_M(x, y, z)$, clearly $\sum_3^0$, while the second and last are diophantine from (2.3) and (2.4) of [2]. The remaining conjunct, $y = P_l(\mathbf{n})$, is easily seen from (2.3), (2.4), and (2.5) of [2] to be of the form

$$\bigvee_{y_1} \cdots \bigvee_{y_l} (y_1 = P(n_1) \wedge \cdots \wedge y_l = P(n_l) \wedge C)$$

where C is the diophantine part remaining after removal of all $P(n_i)$, $1 \leqslant i \leqslant l$. Each $y_i = P(n_i)$ is $\sum_3^0$ with sole $\bigwedge$-quantifier $\bigwedge_{j_i}^{|n_i|}$ (see [1, p. 266]). Thus, the logical form of $\mathrm{ACC}_M(\mathbf{n})$ is

$$\bigvee_x \bigvee_y \bigvee_z \bigvee \cdots \bigvee \left\{ \bigwedge_k^{\beta(|\mathbf{n}|)} A(k) \wedge \bigwedge_{j_1}^{|n_1|} G_1(j_1) \wedge \cdots \wedge \bigwedge_{j_l}^{|n_l|} G_l(j_l) \wedge D \right\}. \tag{3}$$

Variable k occurs only in A and each j_i only in G_i, and A, $G_1, \ldots, G_l$, D are all diophantine. In form (3), therefore, all $\bigwedge$-quantifiers can be factored en bloc, and

the result is $\sum_3^0$. Note that, up to now, no quantifier bounding function has entered the matrix. This concludes our sketch of the removal of Lemma 2.3.

Finally, in [2] we used Lemma 2.2 to collapse adjacent $\bigwedge$-quantifiers into a singleton. The reader will note that, at this point, the time-bounding function β entered the matrix of the EEBA form, for the *first and only* time. We can quickly sketch how to avoid this step. Using condition (i) of Definition 1 in combination with the conjunctive form of the matrix of (3), it is easily seen that predicate $\mathrm{ACC}_M(\mathbf{n})$ can be equivalently expressed as:

$$\bigvee \cdots \bigvee \bigwedge_k^{\beta(|\mathbf{n}|)} \{A(k) \wedge [k > |n_1| \vee G_1(k)] \wedge \cdots \wedge [k > |n_l| \vee G_l(k)] \wedge D\}. \quad (4)$$

If we again remove the predicates $k > |n_i|$, using Adleman's diophantine characterization of exponentiation, the matrix of (4) becomes diophantine and we have

LEMMA 1. *The predicate* $\mathrm{ACC}_M(\mathbf{n})$ *is* $\sum_3^0$-EEBA. *The sole* $\bigwedge$*-quantifier is bounded by* $\beta(|\mathbf{n}|)$, *and* β *does not appear in the matrix.*

Thus, if β belongs to a polynomially closed class of bounds, B, and if R is an l-ary relation accepted by a non-deterministic Turing machine, M, in time bounded by $\beta(|\mathbf{n}|)$, then R has a "normal form" representation of the form

$$\bigvee_y^{2^{p(\beta,|\mathbf{n}|)}} \bigwedge_k^{\beta(|\mathbf{n}|)} \bigvee_{v_1}^{2^{q_1(\beta,|\mathbf{n}|)}} \cdots \bigvee_{v_m}^{2^{q_m(\beta,|\mathbf{n}|)}} \{Q_R(\mathbf{n}, y, k, v_1, \ldots, v_m) = 0\} \quad (5)$$

where Q_R is a polynomial depending on the accepting machine, M, but independent of β, and p, $q_1, \ldots, q_m$ are polynomial functionals of one function argument, β, and l number arguments. (Notations of the type $p(\beta, |\mathbf{n}|)$ are a shorthand for $p(\beta(|\mathbf{n}|), |\mathbf{n}|)$.)

If we use the natural designation *NB* for the class of relations accepted by a NDTM in time bounded by a member of B (the B-analogue of NP), we may restate the Normal Form theorem of [2] as

THEOREM 1. *NB is precisely the class of relations, R, possessing representations of the form* (5).

The only remaining part in the proof of Theorem 1 is that relations of form (5) lie in *NB*. This is quite clear, using the polynomial closure of the class B (condition (ii) of Definition 1).

In comparing Jukna's paper [3] with the above, one finds his Theorem 3.3, which is equivalent to our Theorem 1 if B is taken to be the class, π, of polynomials, or one of the classes

$$2^{\pi}, 2^{2^{\pi}}, 2^{2^{2^{\pi}}}, \ldots.$$

Jukna further shows that the infinite union of the classes of sets so represented, by one of his permissable choices for B, is the third Grzegorczyk class, or the Kalmar elementary sets.

3. Applications: Complexity Hierarchies and Uniform Normal Form

We may use the improved normal form results of this paper in two immediate applications. The first is to unite certain "complexity hierarchies" found in the literature. Like the arithmetical hierarchy of Kleene [4], these hierarchies of recursive sets are defined by (possibly) ascending sequences of sets of recursive sets where, for each level, k, the defining predicates for set membership are of two prenex normal form types. The hierarchies we wish to unite are the *polynomial time hierarchy* of Meyer and Stockmeyer [5], the *diophantine hierarchy* of Adleman and Manders [6], and the *polynomial matrix hierarchy*, as defined in [2]. The three hierarchies are united by the form of the quantifier prefixes of the set definitions of level k. These definitions are

$$n \in S \Leftrightarrow \bigvee \bigwedge \cdots \bigotimes M \qquad \text{or} \qquad n \in S \Leftrightarrow \bigwedge \bigvee \cdots \bigotimes M.$$

Each quantifier symbol represents a block of quantifiers of that type; there are k alternations of type, and the rightmost is determined by the lead symbol and the parity of k. Each individual quantifier is bounded above by some function $2^{\beta(|n|)}$ with possibly different functions $\beta \in B$. In the original sources [2, 5, 6], B is always the set of polynomials, but we allow classes B satisfying Definition 1.

The difference between the several hierarchies lies in the nature of the matrix, M, of the prenex form. For the polynomial time hierarchy, it is any recursive predicate $M(n, \mathbf{w})$, where $\mathbf{w}$ is the sequence of quantified variables, and where the value of M is deterministically computable in time bounded by $\beta(|n|, |\mathbf{w}|)$, for $\beta \in B$. For the diophantine hierarchy, M is a diophantine predicate of n, $\mathbf{w}$ whose internal $\bigvee$-quantifiers are bounded as those in the prefix. For the polynomial matrix hierarchy, the matrix is a polynomial in n, $\mathbf{w}$. It is clear that level-k diophantine hierarchy sets belong to level k, or $k+1$, in the polynomial matrix hierarchy, while level-k sets of both of these belong also to level k of the polynomial time hierarchy. Theorem 1 can then be used to close the hierarchy containments by showing that level-k sets of the polynomial time hierarchy are contained among level-$(k+3)$ sets of the polynomial matrix hierarchy. The proof follows lines similar to Lemmas 3.1 and 3.2 of [2], so we omit details.

THEOREM 2. *The quantifier hierarchies described above, with quantifier bounds in a polynomially closed class B of time bounds, are coextensive.*

The method of the proof of Theorem 1 can also be applied to gain an arithmetical representation of the operations of a universal Turing machine, of l-

arguments. This yields a single arithmetical form (an arithmetical functional with function argument $\beta \in B$) which simultaneously represents all l-placed relations in NB. Such a "universal normal form" for non-deterministic time-bounded acceptability is possible because the time-bounding function, β, does not affect the form of the polynomial matrix, and can be entered as a simple function argument in the quantifier bound functionals (while its index, c, may be an argument in the polynomial matrix).

We visualize a universal Turing machine, $\mathscr{U}$, initialized on its first work tape by three strings of binary integers. The leftmost is of length not exceeding $\beta(|\mathbf{n}|)$ and is the "guess string" used by the TM to decide between its various (non-deterministic) instructions, in the sense of the Garey and Johnson NDTM's (see [7]) we have used in [1, 2]. The middle string gives the binary code, m, of the TM to be simulated. The rightmost contains the binary code for inputs $n_1, \ldots, n_l$. A second work tape is initialized with a binary code, c, for the time bounding function, β, from which, according to condition (iii) of Definition 1, $\beta(|\mathbf{n}|)$ can be computed in time $O(\beta(|\mathbf{n}|))$, and thus by the universal machine in time $O(|c| \times \beta(|\mathbf{n}|))$. Once the "value of the clock," $\beta(|\mathbf{n}|)$, is computed on the second work tape, the clocked simulation of machine m, on input $\mathbf{n}$, can be carried out in time $O(|\mathbf{n}| + \beta(|\mathbf{n}|) \times |m|)$, using the method of simulation in linear time given by Fürer (see Lemma, Sect. 4, of [8]). The running time of the universal machine in performing this simulation is a simple polynomial functional $U(\beta, |\mathbf{n}|, |m|, |c|)$ (even with quadratic slowdown returning to a 1-tape machine). The need to "clock" the simulation by the universal TM may not be entirely obvious since the considerations above make it clear that an unclocked simulation could be accomplished in a readily derived time bound polynomial in β. However, if the universal machine were allowed to run for a time given by a generally valid bounding function, it might simulate more computations that machine m can do in time $\beta(|\mathbf{n}|)$, and so affect the result of the non-deterministic calculation.

If we consider the set of $l+2$ tuples $(n_1, \ldots, n_l, m, c)$ accepted by the universal machine, $\mathscr{U}$, in time $U(\beta, |\mathbf{n}|, |m|, |c|)$, Theorem 1 yields the following result.

THEOREM 3. *There is a* single *arithmetical form* $\mathscr{F}_B(\beta, \mathbf{n}, m, c)$ *so that:* NDTM, m, *accepts* $\mathbf{n}$ *in time* $\beta(|\mathbf{n}|)$ *if, and only if,* $\mathscr{F}_B(\beta, \mathbf{n}, m, c)$. *The form* $\mathscr{F}_B(\beta, \mathbf{n}, m, c)$ *is given by:*

$$\bigvee_{v}^{2^{P(\beta,|\mathbf{n}|,|m|,|c|)}} \bigwedge_{k}^{U(\beta,|\mathbf{n}|,|m|,|c|)} \bigvee_{u_1}^{2^{Q_1(\beta,\ldots)}} \cdots \bigvee_{u_\lambda}^{2^{Q_\lambda(\beta,\ldots)}} \{Q_U(\mathbf{n}, m, c, v, k, \mathbf{u}) = 0\}$$

where P, U, $Q_1, \ldots, Q_\lambda$ *are fixed polynomial functionals of one function argument and* $l+2$ *number arguments, and* Q_U *is a fixed polynomial* (*which does not involve the time bounding function* β).

It is clear that Theorem 3 provides a uniform form, with a fixed number of quantifiers, which characterizes B-bounded NDTM acceptable sets of l-tuples. Corollary

3.3 of [3] obtains a form involving a fixed number of quantifiers, but restricted to the classes B for which its Theorem 3.3 is valid, and not otherwise stating universality.

REFERENCES

1. C. F. KENT AND B. R. HODGSON, An arithmetical characterization of NP, *Theoret. Comput. Sci.* **21** (1982), 255–267.
2. B. R. HODGSON AND C. F. KENT, A normal form for arithmetical representation of NP-sets, *J. Comput. System Sci.* **27** (1983), 378–388.
3. S. JUKNA, Arithmetical representations of machine complexity classes, *Mat. Logika Primenen.* **2** (1982), 92–107. [Russian]
4. S. C. KLEENE, "Introduction to Metamathematics," Van Nostrand, Princeton, N.J., 1952.
5. L. STOCKMEYER, The polynomial-time hierarchy, *Theoret. Comput. Sci.* **3** (1977), 1–22.
6. L. ADLEMAN AND K. MANDERS, Diophantine complexity, *in* "Proceedings 17th Annual Symposium on Foundations of Computer Science," pp. 81–88, IEEE, New York, 1976.
7. M. GAREY AND D. JOHNSON, "Computers and Intractability: A Guide to the Theory of *NP*-Completeness," Freeman, San Francisco, 1979.
8. M. FÜRER, Data structures for distributed counting, *J. Comput. System Sci.* **28** (1984), 231–243.